A NOVEL SECURE H.264 TRANSCODER USING SELECTIVE ENCRYPTION

Nithin M Thomas, Damien Lefol, David R Bull, David Redmill

Department of Electrical and Electronic Engineering
University of Bristol, UK

ABSTRACT

In digital broadcast TV systems, video data is normally encrypted before transmission. For in-home redistribution, it is often necessary to transcode the bitstream to achieve optimum utilization of available bandwidth. If a signal is decrypted before transcoding and re-encrypted, this may lead to a security loophole. This paper presents a solution in the form of a novel H.264 selective encryption algorithm that encrypts sign bits of transform coefficients and motion vectors to allow secure transcoding without decryption. The performance of this system is compared with I-frame encryption. The results show that sign encryption is more secure than I-frame encryption and has a lower complexity. A hybrid system using a modified transcoder and sign encryption is found to give an optimal compromise between security and transcoding performance.

Index Terms— Video Coding, TV Broadcasting, Security, Cryptography

1. INTRODUCTION

Recent advances in multimedia technologies have led to a growth in the varieties of devices capable of handling digital video data. The subsequent ease of unauthorized copying and distribution of the data has led to various copy protection strategies and more generally Digital Rights Management. The challenges that followed in efficient storage and distribution of this data has attracted much interest in the area of video transcoding and scalable coding. While scalable coding can provide efficient solutions for bit-rate reduction by truncating the bitstream, the additional functionality offered by transcoding such as syntax conversion are not supported. Scalable coding also adds significant syntax overhead to the bitstream. When dealing with legacy systems like the ones commonly found in television broadcasting equipment, the video is often not coded in a scalable manner. For in-home redistribution of broadcast video, it is often preferable to bit-rate transcode the data to achieve optimum utilization of available bandwidth. This is often not possible due to the constraints of content security. An encrypted video stream must be decrypted at the transcoder in order to allow transcoding. The data is then re-encrypted before transmission as shown in Fig. 1. This allows efficient and secure distribution of the data, assuming the transcoder is a trusted and tamper-proof device. Devices such as the Secure Video Processor [2] can be used to ensure that the transcoder cannot be tampered with. This increases the cost of manufacturing transcoders, which is vital in consumer applications. A transcoding architecture that has security inherently built into it would therefore be preferable. A secure system for distributing scalable H.264 data is presented in [1]. Little research has been done in integrating protection strategies into transcoder architectures to allow secure transcoding of data.

The H.264 video coding standard [3] has attracted much interest from content providers due to its versatility and coding efficiency. This paper presents a novel H.264 selective encryption algorithm that allows bit-rate transcoding to be carried out on the encrypted bitstream without decryption. The security of the encryption and the performance of the transcoder are compared with a system presented in [4], which encrypts only residuals of I-frames.

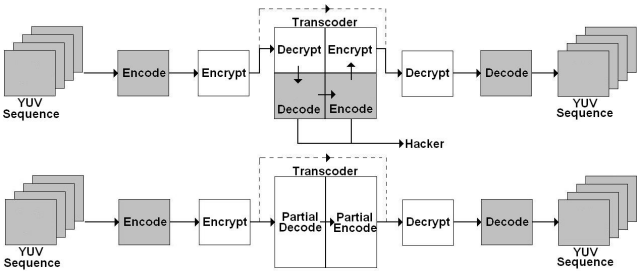

Fig. 1 - Traditional secure transcoding system (top), secure transcoding system (bottom)

2. SELECTIVE ENCRYPTION

Various approaches are presented in the literature that support encryption of portions of video bitstreams to reduce computational overhead. To allow transcoding without decryption, the encryption strategy must preserve parts of the bitstream used by the transcoder. Encryption of I-frames and header data of I-blocks is presented in [5]. This approach is extended in [6] to encrypt

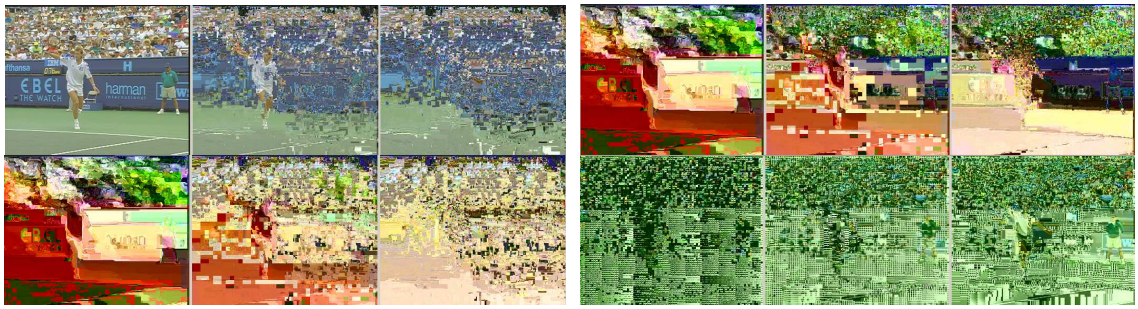

Fig. 2 - I (left), B (middle) and P (right) frames using encryption on Motion Vector Sign Bit (top left), DCT Sign Bit (top right), Motion Vector and DCT Sign Bit (bottom left) and I-frame (bottom right)

selected I-macroblocks and their headers. Encryption of DCT coefficients is presented in [7]. Encrypting only the sign bits of DCT coefficients is discussed in [8] while a similar approach on motion vectors is used in [9].

When choosing the most appropriate encryption method for use with a transcoder, it is important to consider two aspects of the algorithm:-

- The portions of the bitstream that are used in the transcoding must not be affected by encryption,
- The encrypted video must not be intelligible without knowledge of the decryption key.

The first criterion is important in order to achieve the highest possible performance from the transcoder. In order to fulfill this requirement, all header data must be available to the transcoder and therefore cannot be encrypted. The DCT coefficients and motion vectors are requantized and sometimes refined by the transcoder, therefore they cannot be encrypted either. One solution is this to transcode only the P and B-frames using a transform-domain transcoder. This allows encryption of the I-frame intra-prediction residuals as described in [4]. Alternatively, encrypting only the sign bits of DCT coefficients and motion vectors allows the transcoder to carry out requantization.

The intelligibility criterion is important in order to maintain the security of the system. The amount of information available about the plaintext video from the ciphertext is indicative of the level of security and hence the probability of carrying out a successful attack on the system. Fig. 2 shows frames encrypted using different schemes and decoded without decryption.

The encryption on the motion vectors was carried out using the RC4 stream cipher [10] that generates a random sequence of keystream bytes. If the value of a byte corresponding to a motion vector is even, then the sign of the motion vector is negated. If the value of the byte is odd, the sign is left unchanged. Using a stream cipher to encrypt the transform coefficients caused synchronization problems in the decryption stage as the transcoder sets some of the coefficients to zero during requantization. These zero coefficients caused a loss of synchronization between the keystream and the ciphertext. A block cipher was therefore used to encrypt the transform coefficients. The macroblock and block address of the coefficient to be encrypted were encrypted using Rijndael [10]. If the value of the resulting ciphertext was even, the sign of the transform coefficient was negated. When using I-frame encryption, the residual data in the I-frames were also encrypted using Rijndael. For coding parameters of the input sequence, the reader is referred to Section 4.

Encrypting the motion vectors or DCT coefficients alone clearly provides insufficient security due to some content being visible as seen in Fig. 2. When the DCT coefficients and motion vectors are both encrypted together, the intelligibility of the bitstream is severely degraded. Although the I-frames still reveal some edge information about the plaintext, when the sequence is played, it appears mostly as random noise. In Fig. 2 (d) the I-frame encryption reveals some information in the P-frames. On close inspection, the player and linesman can both be seen. When the sequence is played, the contents of the scene can be seen even more clearly and even small objects such as the tennis ball can actually be followed. Although not apparent from the figure, during playback, encryption of the I-frames provides far less degradation than encrypting the sign bits. This is because of the I-macroblocks in these frames that are not encrypted. I-frame encryption and sign encryption can be integrated with the transcoders as shown in Fig. 3.

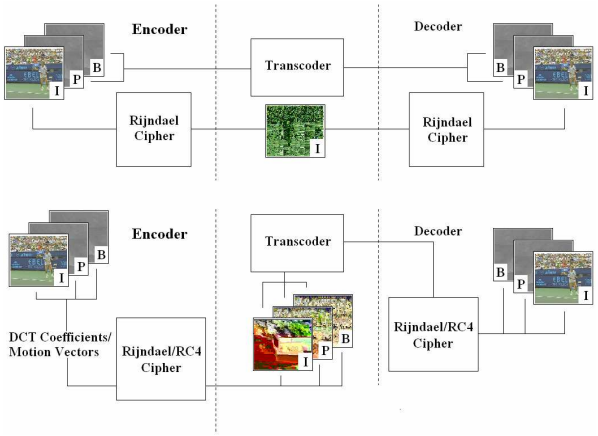

Fig. 3 - I-frame (top) and sign bit encryption (bottom)

3. TRANSCODING

Various transcoding architectures for coded video are described in [11] and [12]. The Cascaded Pixel Domain architecture is shown to produce the most optimal R-D curve. The CPDT approach however requires the frames to be transformed into the pixel domain. This is not possible on an encrypted bitstream as the pixel values are dependant on the data that is encrypted. The Fast Pixel Domain Architecture carries out transcoding using the transform coefficients and motion vectors in the transform domain and so does not require knowledge of the pixel values. This system requantizes the coefficients and carries out error corrections on the residuals to compensate for any drift that may be introduced due to the transcoding process. An open loop architecture can also be used that merely carries out requantization in order to reduce the bitrate.

The closed loop FPDT generally gives better performance than the open loop system [12]. I-frame encryption and sign bit encryption were both used with the closed and open loop architectures. Both systems had to be modified to deal with I-frame encryption. Transcoding the encrypted I-frames would make the data undecipherable at the decoder. This means that the I-frame data has to be copied, without modification, into the output of the transcoder. The P and B frames however were transcoded as normal. The modified system is referred to as the Inter-frame transcoder from here on.

4. RESULTS

All simulations were carried out on the 'Stefan' sequence coded at 10 Mbps at CIF resolution. One in every 30 frames was coded as an I-frame. Fig. 4 shows the performance of the open loop and closed loop systems using sign bit encryption.

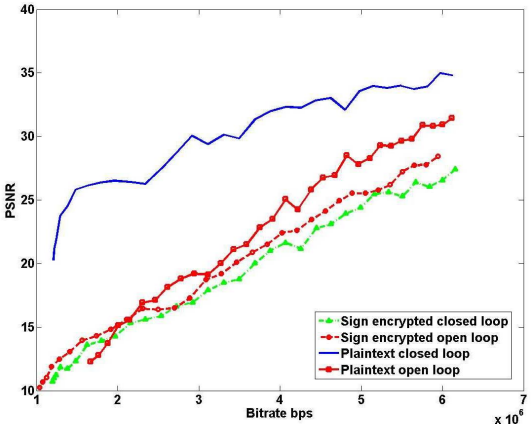

Fig. 4 - Performance of transcoders with sign bit encryption.

When sign encryption is used, the performance of the transcoder is around 5 dB lower for the open loop system and around 13 dB lower for the closed loop system when compared with no encryption. This deterioration is caused by errors introduced by the transcoder. In the closed loop system, there is a drastic effect on the transcoder due to the transform coefficients of the prediction residuals being encrypted. In a system with no encryption, the closed loop improves the quality of residuals by compensating for any errors introduced by transcoding. When the transcoder operates on encrypted bitstreams, the actual residuals are not known to the transcoder. This means that the compensation carried out to the residuals by the transcoder actually leads to a reduction in the accuracy of the residuals. In the case of both the open and the closed loop systems, encrypting the sign bits of the transform coefficients alters the statistics of the VLC code-words. This change causes some of the code-words to be decoded incorrectly. These errors reduce the performance of the closed loop system further and cause the 5 dB drop in the open loop system.

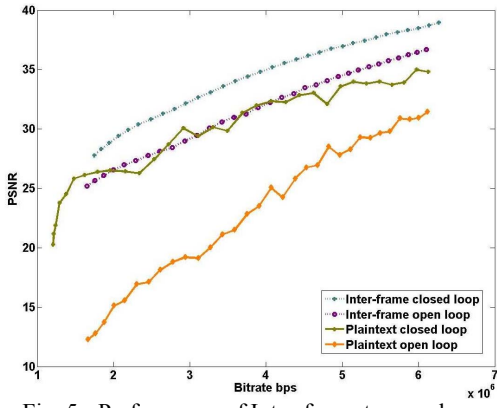

Fig. 5 - Performance of Inter-frame transcoders.

The Inter-frame transcoders perform around 4 dB better and between 4 dB and 13 dB better for the closed and open loop systems respectively as shown in Fig. 5. This is because the quality of the I-frames is preserved during transcoding. When the sequence is decoded, the predictions are therefore made using references of a higher quality. If the I-frames are transcoded, the quality of the residuals deteriorates, leading to a lower performance.

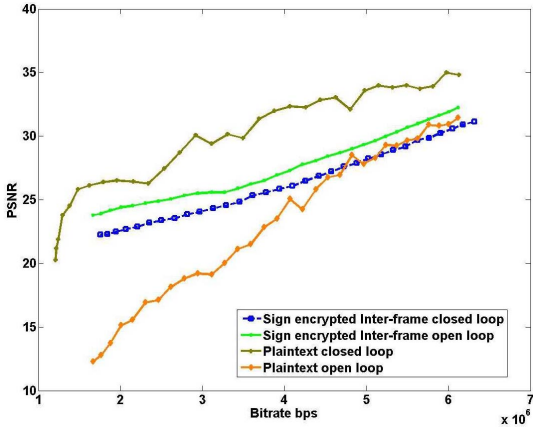

Fig. 6 – Inter-frame transcoders with sign encrypted data

The gain in performance by using sign encryption with the Inter-frame transcoder compensates for losses introduced by the encryption. The open loop system performs 2 to 10 dB better than the unmodified plaintext system as shown in Fig. 6. The closed loop system performs between 3 to 7 dB lower than the plaintext transcoder. This difference is due to a large number of residual errors introduced by the transcoder. Fig. 7 shows sequences produced by transcoding encrypted and plaintext sequences.

Fig. 7 – Transcoded, decrypted frames: plaintext (top left), sign encrypted (top right), inter-frame transcoder + plaintext (bottom left), inter-frame transcoder + sign encrypted (bottom right)

In many applications, the complexity of encryption may need to be considered due to real time processing requirements. I-frame encryption encrypts the largest volume of data and is therefore the slowest. This however is not true for the transcoder. 300 frames of the 'Stefan' sequence, as described above, were transcoded from 10 to 3 Mbps. The time taken to transcode the sequence was 4% faster with the Inter-frame transcoder than the unmodified version. I-frames processing takes up a large portion of the resources, so eliminating this leads to a faster transcoder. The sign encryption does not affect the transcoder speed.

5. CONCLUSIONS

The comparison between I-frame encryption and sign bit encryption has shown that the choice of encryption algorithm is dependant on the characteristics of the application. For a system that demands complete unintelligibility, I-frame encryption does not provide sufficient security. The information leaked from this system also suggests that it may be possible to carry out a high level attack that uses the characteristics of the video together with the information leaked from the encrypted data to re-construct most of the sequence. The sign bit encryption however leaks much less information.

Both encryption schemes affect transcoder performance. I-frame encryption required the standard transcoder to be modified in order to cope with the encrypted data. This modification produced higher PSNR for all bitrates. The Sign bit encryption scheme reduced the PSNR of the transcoded sequence due to errors introduced by the transcoder. Performance is drastically improved when sign bit encryption is used with Inter-frame transcoders.

For applications where the quality of the transcoded sequence is paramount, such as DVB, I-frame encryption may be more suitable. For systems such as TV on mobile phones, where the quality of the sequence is not the most important factor in deciding the security protocols, sign bit encryption would be more appropriate. For real time systems that demand high speed processing, the complexity of I-frame encryption may prove too high due to the large volume of data encrypted. Sign encryption on the other hand encrypts less data and therefore adds much less overhead.

The hybrid system comprising of the Inter-frame transcoder and sign bit encryption is found to provide a good compromise between security and transcoding performance.

6. REFERENCES

[1] S. Wee, and G. Apostopoulos, "Secure Scalable Video Streaming for Wireless Networks", *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, Vol. 4, pp. 2049 – 2052, 2001

[2] *SVP Open Content Protection System; Technical Overview* (http://www.svpalliance.org/resources.hmtl)

[3] Draft ITU-T Recommendation and Final Draft International Standard Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC), Mar. 2003.

[4] N. Thomas, D. Lefol, D. Bull and D. Redmill, "Transcoding Selectively Encrypted H.264 Bitstreams", *Proc. 25th Int. Conf. on Consumer Electronics* Jan. 2007.

[5] J. Meyer, and F. Gadegast, "Security Mechanisms for Multimedia Data with the Example MPEG-1 Video", *Project Description of SECMPEG*, May 1995.

[6] M. Alattar, I. Al-Regib, A. Al-Semari, "Improved Selective Encryption Techniques for Secure Transmission of MPEG Video Bit-Streams", *Proc. 1999 Int. Conf. on Image Processing*, Vol. 4, pp. 256-260, Oct. 1999.

[7] L. Tang, "Methods for Encrypting and Decrypting MPEG Video Data Efficiently", *Proc. 4th ACM Int. Multimedia Conf.*, pp. 219-230, Nov. 1996.

[8] C. Shi and B. Bhargava, "A Fast MPEG Video Encryption Algorithm", *Proc. 6th Int. Multimedia Conf.*, Sept. pp. 81 – 88, 1998

[9] C. Shi, Y. Wang, and B. Bhargava, "MPEG Video Encryption in Real-Time Using Secret Key Cryptography", *Int. Conf. on Parallel and Distributed Processing Techniques and Applications*, June 1999.

[10]N. Smart, *Cryptography an Introduction*, pp 102-106, McGraw-Hill 2003.

[11]A. Vetro, C. Christopoulos, and H. Sun, "Video Transcoding Architectures and Techniques: An Overview", *IEEE Signal Processing Magazine*, pp. 18 – 29, March 2003.

[12]D. Lefol, D. Bull, and N. Canagarajah, "Performance Evaluation of Transcoding Algorithms for H.264", *IEEE Trans. on Consumer Electronics*, Vol. 52, No. 1, February 2006.